PROJECT REPORT

ON
THE MAGGS DIARY

BY

1. Sayyad Muskan (21165)
2. Shinde Shrikant (21173)
3. Powar Ankita (21155)
4. Nagade Ashish (21125)
5. Biradar Gajanan (21028)

SAVITRIBAI PHULE
UNIVERSITY OF PUNE

MASTERS OF COMPUTER APPLICATION
INSTITUTE
PUNE-411019

2022-2023

INDEX

INTRODUCTION:

Since the start of human history, Traveling has a vital importance. No one owns some relatives or property in every corner. But work often leads one to far away locations. Hotels are of prime importance in these and other cases. Hotels constitute of Restaurants, Guest Houses and Resorts. Usually a hotel consists of rooms and Halls to be rented and restaurants on eat-away terms for customers. Big Hotels constitute hundreds of halls, thousands of rooms and dozens of restaurants inside them. Big hotels involve hundreds of employs who serve different posts. Hotels involve waiters, laundry, chefs, hosts, managers and managers of managers on one hand to CEOs on the other.

The management of such a heavy staff and to assign everybody his job is very tough. Management of such a big structure is not a piece of cake. However Software like other fields, serves here as well to ease mankind. Hotel Management Systems provide an easy and effective way to execute the tasks of Hotel Management. All the hotels around the world use different efficient software to help them in their managerial tasks. All big Hotel chains in PAKISTAN also rely on management Software. Management software allows one to have a better connection with his customers as well as presents a better access to people around the world to explore them. On the other hand it also enables one to have a better management and a better control over the employs.

Every big organization relies on some management scheme. Without proper management nothing in this world can go on. To do the managerial tasks properly and effectively one needs highly efficient tools. These tools not only helps one to do his/her work properly but also reduces the chances of error. Because human can forget things, Software cannot. So every organization relies on different database software to perform their respective tasks. In Hotels there are a number of activities going on in a single time. Hundreds of halls, Thousands of rooms are all being used of evacuated at a time. To manage such a large system we need some efficient tool. Hotel management system helps us in this regard. It not only provides the management facility to the organization but also provides the customers and other people to get exact information about the hotel (with the consent of hotel). These systems enable people around the world to make reservations for rooms, hall etc. The management system helps one to check the menus in different restaurants of the hotel and check prices and other information about the hotel. Hotels on the other hand are able to keep record of their each item sold in the restaurant, any room reserved and an accurate guess of the revenue which should be gathered and what is gathered. This helps the owners a lot to keep a check and balance of such a huge system which on the other hand is not possible. The Accounts are managed easily i.e. the salaries of workers and the revenue generated by the customers can be easily managed.

Moreover the most important task, “CUSTOMER SATISFACTION” is truly achieved. As the management provides an efficient access and everything is available to the customer on a single click through the web (for web based systems). Our Hotel Management System also provides satisfactory functionality to all of its users.

Purpose:

The main purpose of the system as described above is better management. This system provides an easy access to the management issues and one can easily get where the relevant person could be found. The world has changed to databases to keep record of everything. The basic need here is to record the data about reservations of room and to calculate the relevant revenue and save its data as well.

OBJECTIVE OF THE PROJECT:

The objectives of our Hotel Management system are as follows:

- Providing better management of ROOM using the DATABASES. This will enable the management to keep a record of their customers and will lessen the chance of any error e.g. the allotment of a single room to more than one customer.
- Enable the customers to get better access to the hotel. Facilitating them to reserve a room for them through distance. This eases customers on one edge and provides business to hotel on the other hand.
- Enabling accounts department to keep record of all reservations and other selling and hence make the fare revenue. The enable the generation of detailed bills for their customers to satisfy their concerns as well. At the best this system enables the CEO and other share holders to check the records and get the revenue estimate; enabling them to find black sheep out if anyone involved.
- Overall a better “Management” of resources, Assigning duty points each day for different workers and assigning everyone his work! This will be done by generating the report showing everyone’s tasks which can be viewed over the system by relevant people.
- Least, enabling general public to know in detail about the hotel, its customs, food and living standards and attractions it can offer.

CONSTRAINTS:

The devised constraints of our system can be to implement the system in such a way that it causes least crashes and the system exceptions are handled properly. The Data

security issue must be handled carefully so that privacy is maintained and no one can access the hotel data from outside.

EXISTING SYSTEM:

- Currently in hotel all the work done manually. When a guest make a reservation, all the reservation details (including guest details) are recorded in a hotel register. At the time of checkout of customer, calculations of bills and inventory items are done manually too.
- Doing all the work manually and storing information on register takes much time and wastes much precious man hours. Manually calculation of bill is also error prone. If management want any old information like room record or reservation details then finding old records is very tiresome task and it takes a lot of time to find records form old files.
- The system allows the manager to post available rooms in the system. The existing system of Hotel Management was manual. All the daily routines is carried out manually and the records are maintained in the record books or the registers.
- Our system keeps these conditions in mind and we provide easy and legal access to everyone around the globe.

PROPOSED SYSTEM:

The basic objective of the hotel management system includes fast access to the data to make decisions in no time. Thus we provide a system which works out the solution in seconds. This enables finding the room reservation records, adding new records and at that time checking correctly that the reservation made is unique.

The system will include the past data about guests and system will maintain the present reservations record as well. The system is designed basically to solve the problems on the issue of reservations and management regarding those reservations. The primary objectives include “CUSTOMER SATISFACTION” and the system must present all the ways to keep the customer happy.

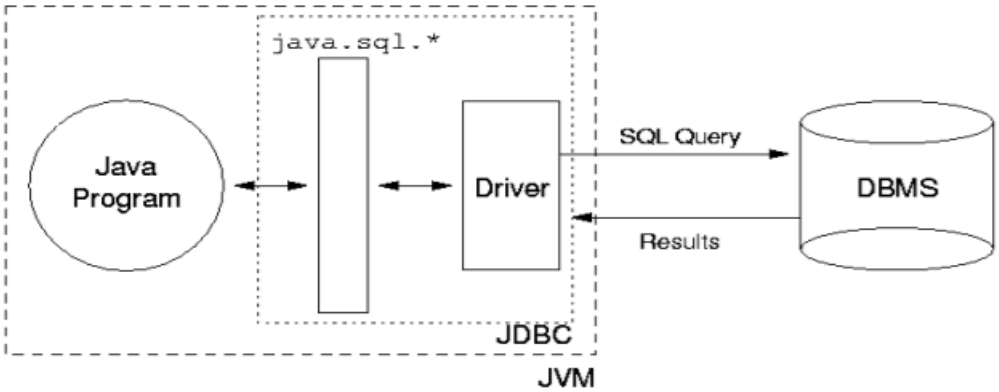
DETAIL DESCRIPTION OF TECHNOLOGY USED:

1. JAVA:

Java is a general-purpose, object-oriented programming language developed by Sun Microsystems of USA in 1991.Originally called Oak by James Gosling (one of the inventor of the language). Java was invented for the development of software for cunsumer electronic devices like TVs, tosters, etc. The main aim had to make java simple, portable and reliable. Java Authors: James , Arthur Van , and others. Java is a high-level, third generation programming language, like C, FORTRAN, Smalltalk, Perl, and many others. You can use Java to write computer applications that play games, store data or do any of the thousands of other things computer software can do. Compared to other programming

languages, Java is most similar to C. However although Java shares much of C's syntax, it is not C. Knowing how to program in C or, better yet, C++, will certainly help you to learn Java more quickly, but you don't need to know C to learn Java. A Java compiler won't compile C code, and most large C programs need to be changed substantially before they can become Java programs. What's most special about Java in relation to other programming languages is that it lets you write special programs called applets ,web project etc. that can be downloaded from the Internet and played safely within a web browser. Java language is called as an Object-Oriented Programming language and before beginning for Java, we have to learn the concept of OOPs(Object-Oriented Programming).

JDBC DRIVER MODEL



In the commercial world, we use Java 2 Enterprise Edition (J2EE) to solve business problems, to develop commercial software, or to provide contract services to other businesses' projects. If a company wants to build an e-business Website using a multitier architecture, it usually involves managers, architects, designers, programmers, testers, and database experts

throughout the development lifecycle

Mapping

Mapping Java classes to database tables is accomplished through the configuration of an XML file or by using Java Annotations. When using an XML file, Hibernate can generate skeletal source code for the persistence classes. This is unnecessary when annotations are used. Hibernate can use the XML file or the annotations to maintain the database schema. Facilities to arrange one-to-many and many-to-many relationships between classes are provided. In addition to managing associations between objects, Hibernate can also manage reflexive associations where an object has a one-to-many relationship with other instances of its own type.

2. MySQL:

MySQL is the world's most used open source relational database management system (RDBMS) as of 2008 that runs as a server providing multi-user access to a number of databases.The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation. MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack (and other 'AMP' stacks). LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python." Free-software-open source projects that require a full-featured database management system often use MySQL.

Interfaces

MySQL is a relational database management system (RDBMS), and ships with no GUI tools to administer MySQL databases or manage data contained within the databases. Users may use the included command line tools, or use MySQL "front-ends", desktop software and web applications that create and manage MySQL databases, build database structures, back up data, inspect status, and work with data records. The official set of MySQL front-end tools, MySQL Workbench is actively developed by Oracle, and is freely available for use.

Graphical

The official MySql Workbench is a free integrated environment developed by MySQL AB, that enables users to graphically administer MySQL databases and visually design database structures. MySQL Workbench replaces the previous package of software, MySQL GUI Tools. Similar to other third-party packages, but still considered the authoritative MySQL front end, MySQL Workbench lets users manage database design & modeling, SQL development (replacing MySQL Query Browser) and Database administration (replacing MySQL Administrator).

MySQL Workbench is available in two editions, the regular free and open source *Community Edition* which may be downloaded from the MySQL website, and the proprietary *Standard Edition* which extends and improves the feature set of the Community Edition.

SYSTEM ANALYSIS:

The system has been designed keeping in mind the user's activities(i.e. manager of the hotel). With the help of this system, the manager can check on the availability of rooms, check in new customers, make changes to a customer's bill based on the services availed by him. Generate a total bill which will have the sum of the bills. The manager will also be able to hire new employees and have a track of their personal and professional details. Through the various validation done in the project we try to ensure that logically wrong entries are not made by the user.

Fact finding methods are used to find out the requirements of the customer which are not in the existing system. A survey was conducted to find out the requirements of the hotel staff who earlier were involved with the system. Lists are now being generated that tell us the name, id and status (checked in/checked out) of each customer. This also helps to find the customer id of a particular customer or vice versa. A similar list is now being prepared of all existing employees and their employee ids.

Must	Should	Could	Wont
Store customer details in database.	Display available and booked rooms	Show number of available rooms for each type of room	Have an interface for customers to make their own bookings
Use sorting algorithm to arrange rooms according to room number	Display add-ons to the complimentary services	Search for customers using name	Login for staff members
Make bookings	Edit bookings i.e. remove already made reservations	Display the bill amount so far before check-out	
Print receipts at checkout			
Remove a booking after check-out and make the room available			

IDENTIFICATION OF NEED:

This involves the collection of requirements which the project must fulfill. This includes the following details.

1. We will require a database system. The database is a major need as it interprets the requirement of storing the data chunks which need to be managed according to proposed standards. The basic need here is speed. As user always requires an efficient and fast system and the other requirement is intact and crash free system. This can run without even a single second of being down! It must fulfill the following requirements:
    - Make Reservations for a Room
    - Cancel a reservation for Room
    - Store a data base for all kinds of reservations
    - Store payment details
    - Store Room and other information.
2. The other major requirement is interface. Interface enables the user to make interactions and usage about the system. The system must present an easy Graphical user interface because most of the users usually are unaware with the use of such systems and complicating them will restrict their usage in proper way this on one hand creates problem for the hotel management and on the other hand is a problem for the developer. The major strongholds kept in this regard are:
    - Allow receptionist to Make/Cancel Reservations
    - Allow receptionist to maintain the guest lists
    - Allow the receptionist to provide the user with bill payments.
    - Allow the HR department to post jobs, look for new employs maintain employ list, maintain guard list for security purpose.
    - Allow the higher managers to keep a look on accounts and other details.
    - Allow the basic information access for general public.

SCOPE OF THE PROJECT:

The scope of our project is to make a management system relevant to Hotels. This management system will serve the need of different people. Anonymous people around the world can get information about the hotel through PEOPLE dedicated section. They are also allowed to book a room for them using e-reservations. The customers on the other hand can book Table, Hall or room and get their bills paid by front desk.

The admin management on the other hand has different sections. Human Resource Department deals in vacancies, Employ relevant problems and security plans. They can set orders and the HR head can access the reports using a single click on his monitor. The Restaurant manager on the other hand manages his menu, waiter's duty and check relevant reservations record on his seat. Similarly Room Division has their maintenance. The Accounts department checks all reservations and other data and maintains the billing section enabling a better and more reliable revenue generating mechanism. While every manager or head of

a department has access to their relevant information, the CEO of the Hotel or other major share holders have access to all the databases to check every entry and position of every person as he desires. He can look through all the details and Revenue guess and actual revenue generated using the software.

Thus a single system serves the need of all the users from general public to the employs of the hotel.

FEASIBILITY STUDY:

1. Technical Feasibility:

The technical aspect of the system is covered with the use of APACHE NETBEANS IDE 12.6. The programming aspects of the systems are designed in java and Jframe works. The GUI is designed by making the usage of Jframe. The data relevant requirement is accomplished by the usage of text files to store and copy the data from. The application on major involves the usage of stream reading/writing and on the GUI side it involves the usage of Buttons/Tab controls and text boxes.

2. Economical Feasibility:

The computerized system will help in automate the selection leading the profits and details of the organization. With this software, the machine and man power utilization are expected to go up by 80-90% approximately. The costs incurred of not creating a system are set to be great, because precious time can be wanted by manually.

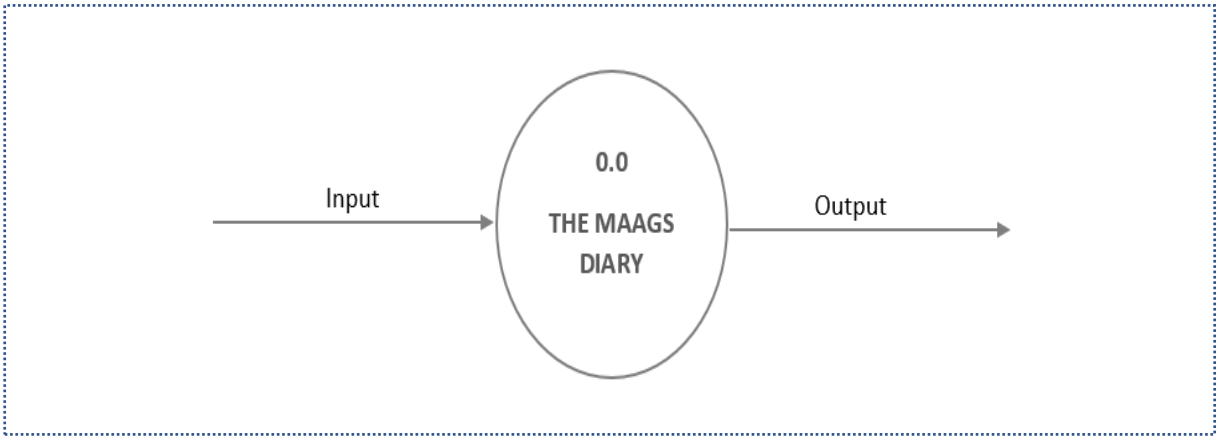
3. Operational Feasibility:

In this project, the management will know the details of each project where he may be presented and the data will be maintained as decentralized and if any inquires for that particular contract can be known as per there requirements and necessaries. Time evaluation is the most important consideration in the development of project. The time schedule required for the developed of this project is very important since more development time effect machine time, cost and cause delay in the development of other systems. A reliable Campus Tracker can be developed in the considerable amount of time.

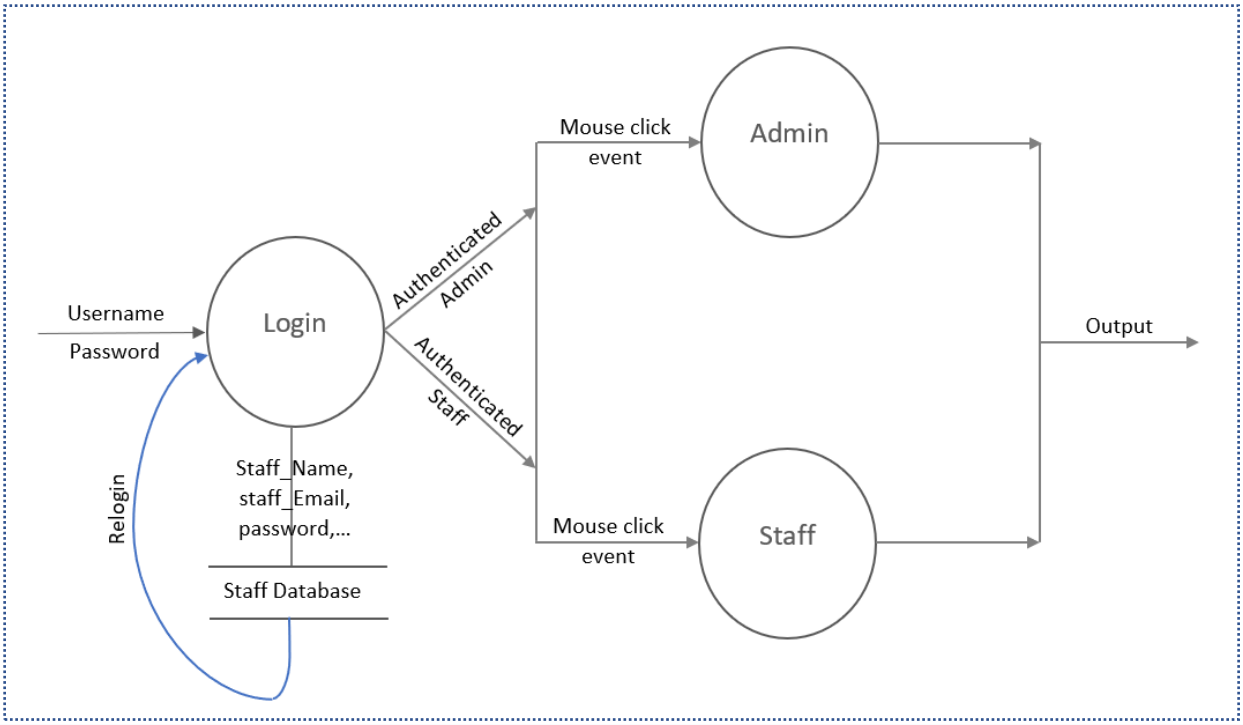
ANALYSIS AND DESIGN:

CONTEXT LEVEL DFD

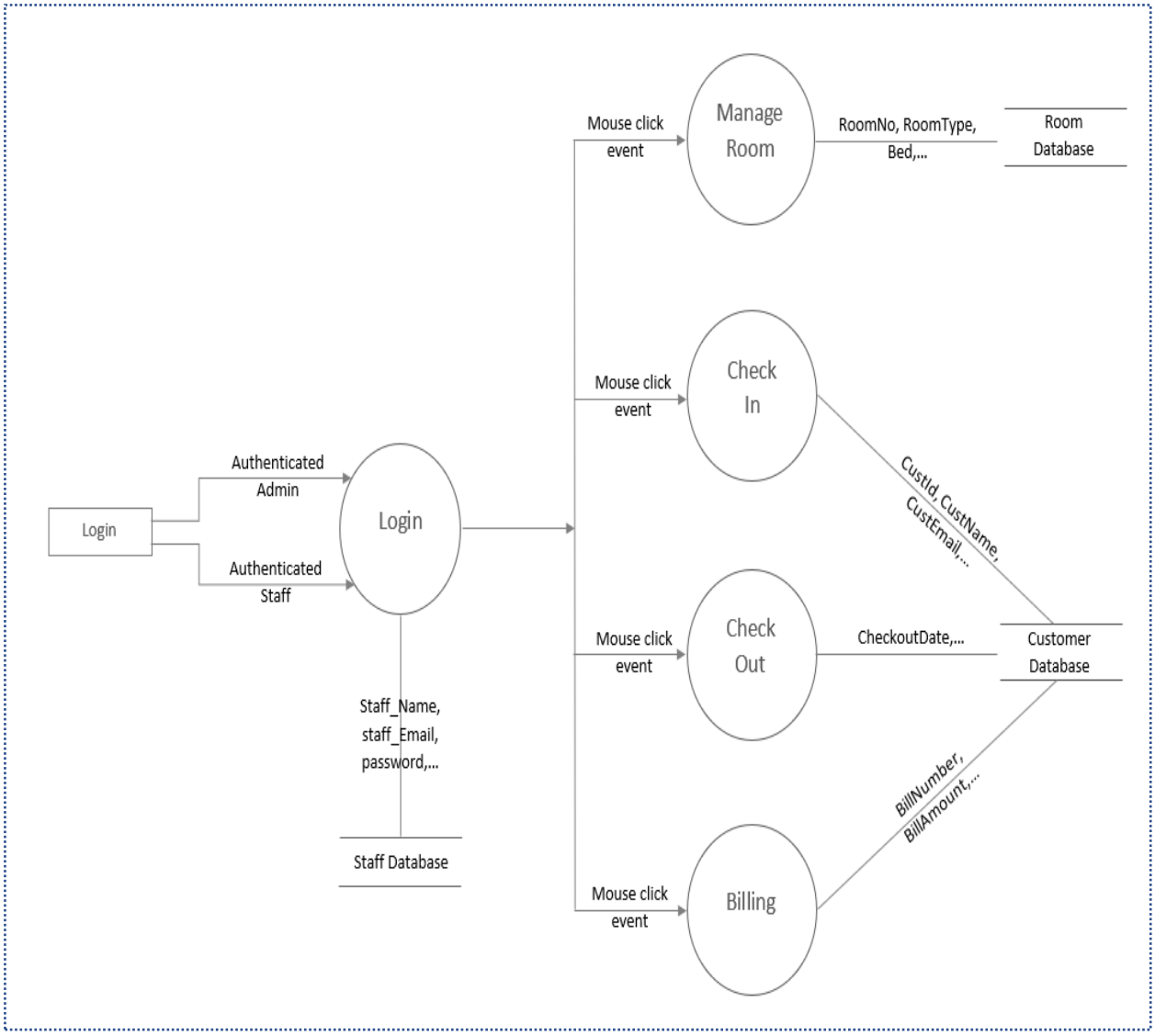
- **Zero level DFD**



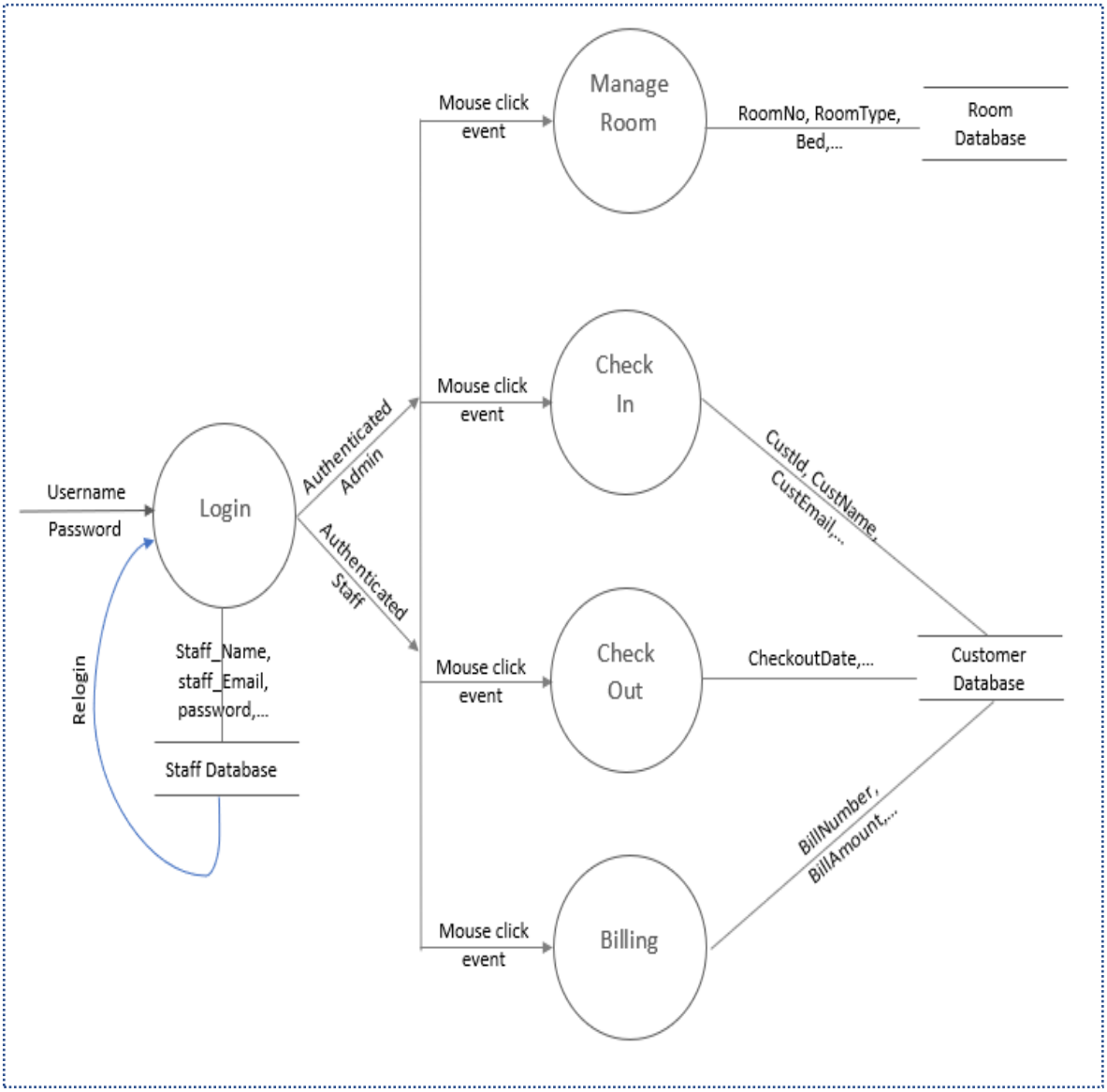
- **First Level DFD**



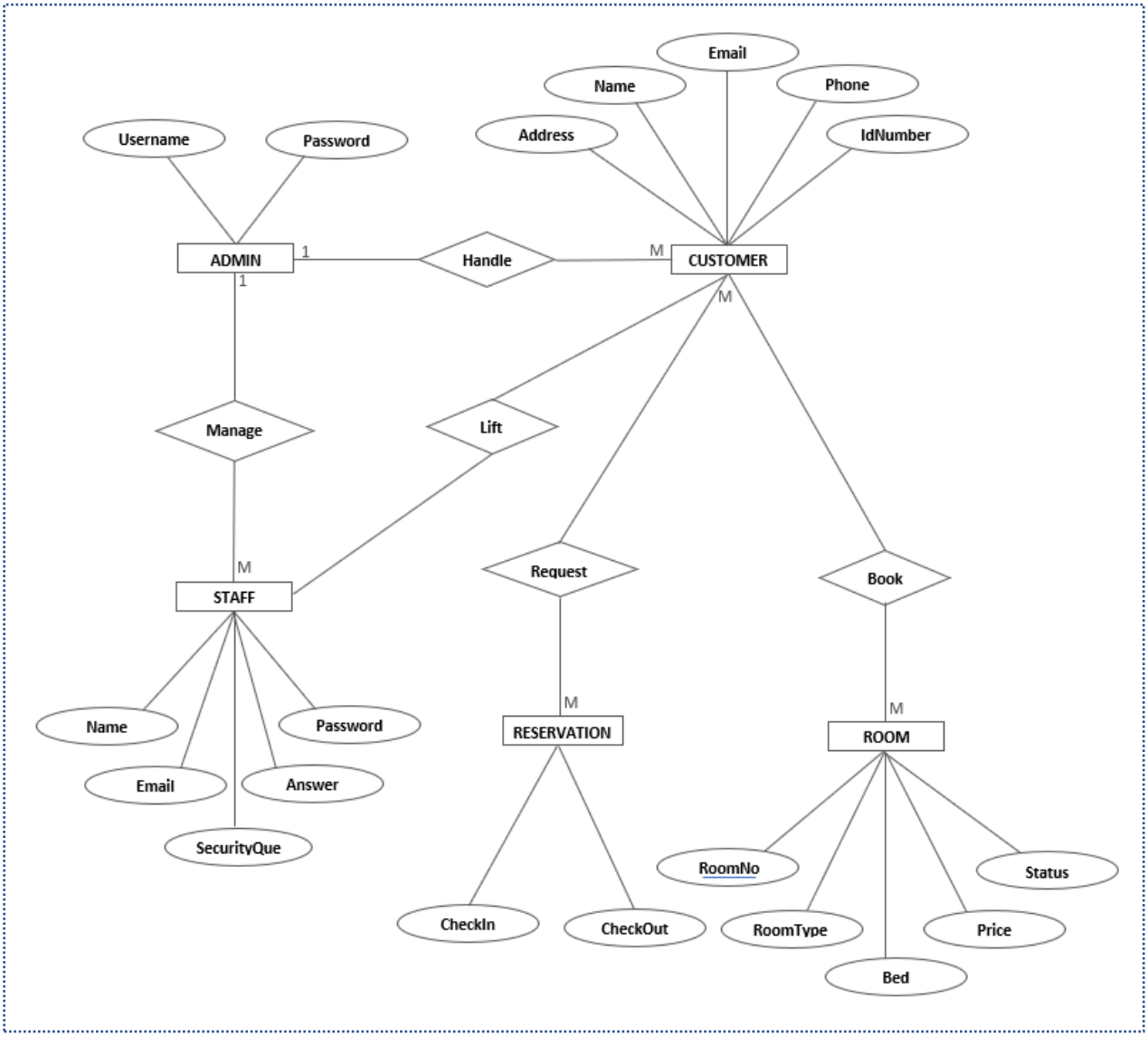
- **Second Level DFD**



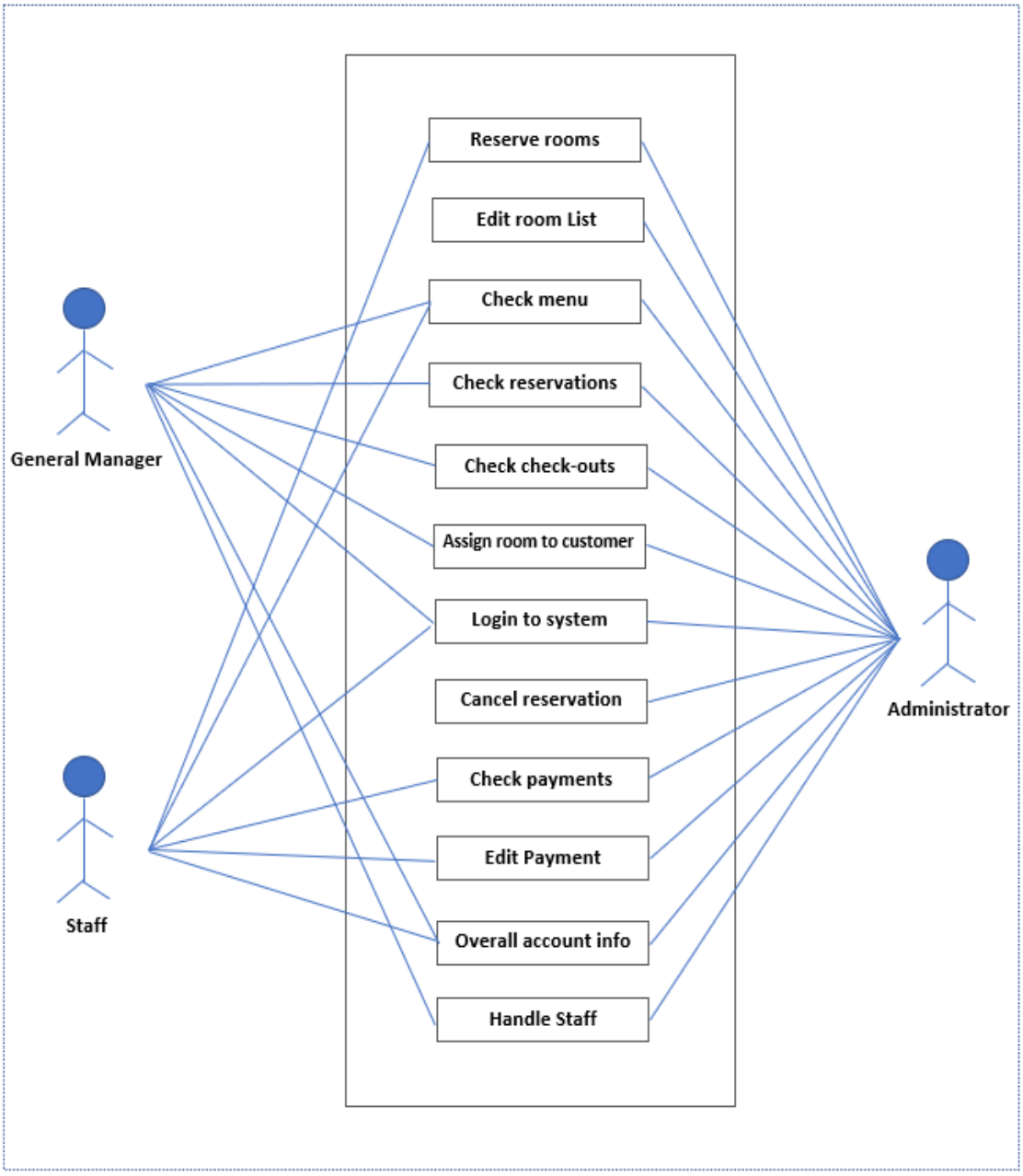
- **Third Level DFD**



- **E-R DIAGRAM**

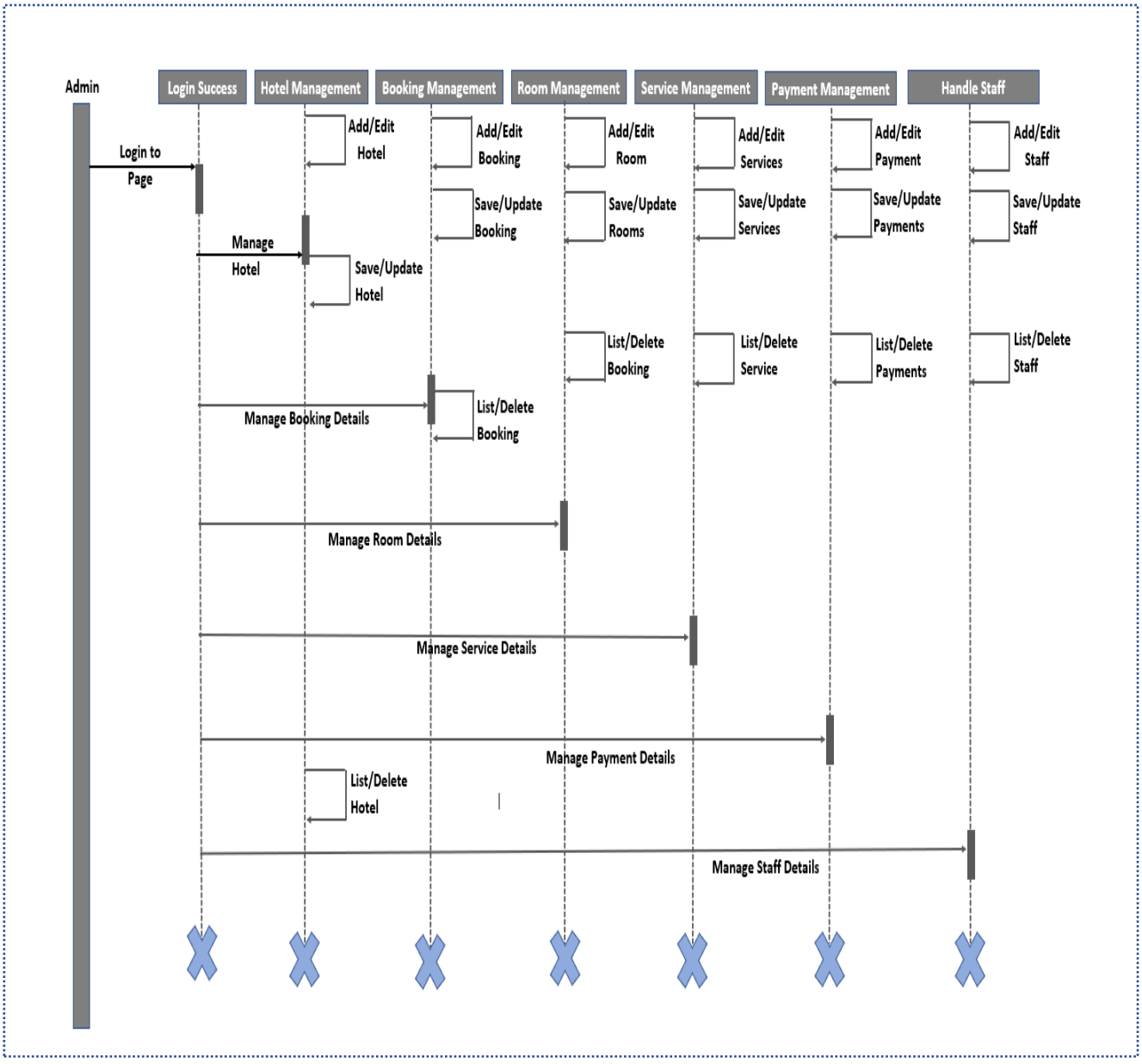

Email
Name
Phone
Username
Password
Address
IdNumber
ADMIN
1
Handle
M
CUSTOMER
1
M
Manage
Lift
M
STAFF
Request
Book
Name
Password
M
RESERVATION
M
ROOM
Email
Answer
SecurityQue
RoomNo
Status
CheckIn
CheckOut
RoomType
Price
Bed

- **USE CASE DIAGRAM**

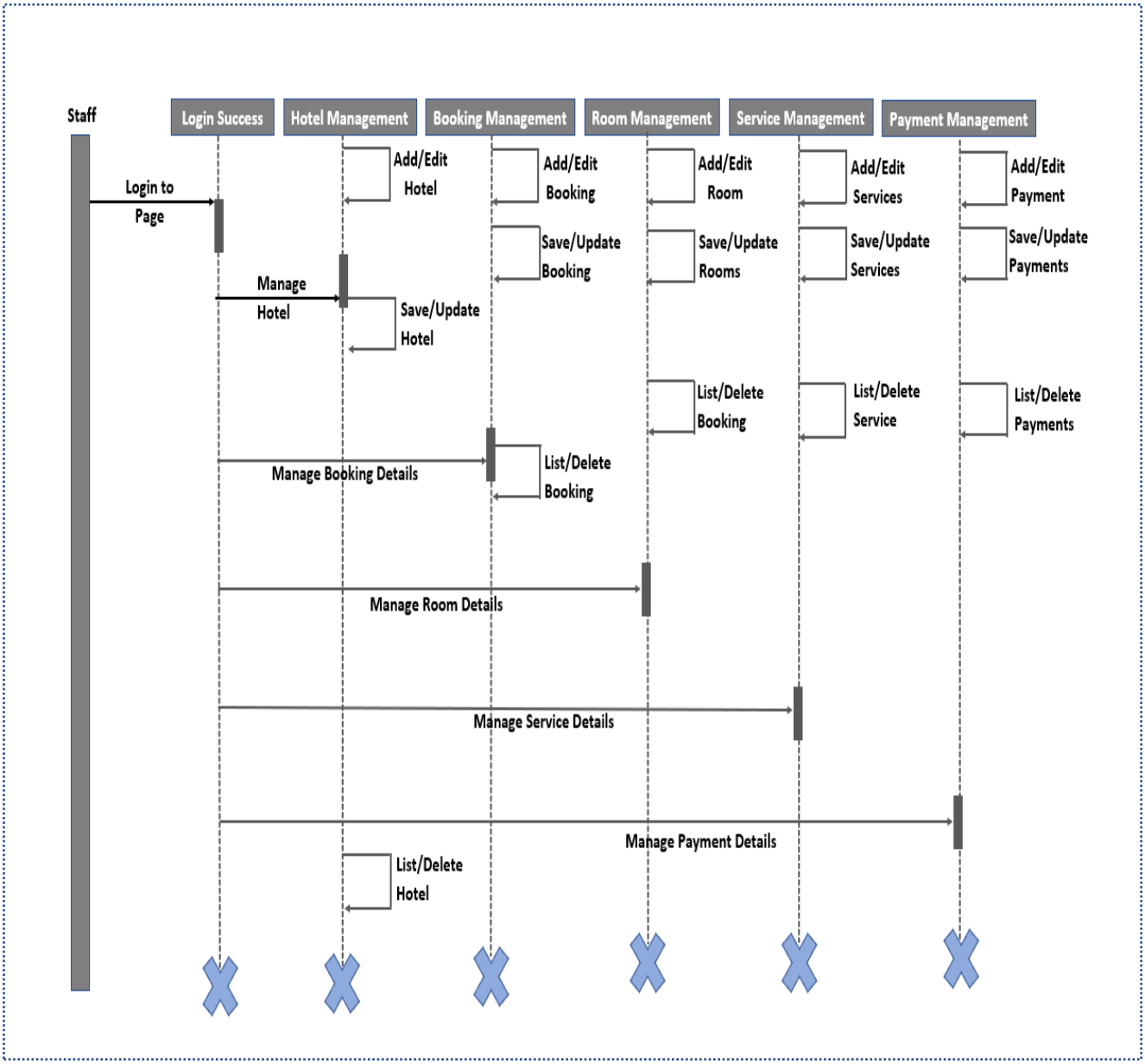


- **SEQUENCE**

  - **ADMIN SECQUENCE DIAGRAM**



➢ STAFF SECQUENCE DIAGRAM



S/W & H/W REQUIREMENT SPECIFICATION

❖ SOFTWARE REQUIREMENT

- **Client Side:**
  Operating System: Windows 10,11
  Web Browser: Google Chrome, Mozilla Firefox, etc.

- **Software (Server Side):**
  Operating System: Windows 10,11
  Front end: JFrame
  Middle ware: Java Development Kit(JDK 1.5.0)
  Backend: MySQL Database Server

❖ HARDWARE REQUIREMENT

- **H/W Configuration (Client Side):**
  Processor –Core i3 & above
  Speed – 2 GHz & above
  RAM – 4 GB & above

- **H/W Configuration (Server Side):**
  Processor – Core i3 & above
  Speed - 2 GHz & above
  RAM – 4 GB & above

SYSTEM DESIGN:

Cardinality - In SQL (Structured Query Language), the term cardinality refers to the uniqueness of data values contained in a particular column (attribute) of a database table.

ROOM DETAILS:

Field	Type	Null	Key	Default
roomNo	int	No	PRI	NULL
roomType	varchar(200)	YES		NULL
Bed	varchar (200)	YES		NULL
Price	int	YES		NULL
Status	varchar(20)	YES		NULL

Keyname	Type	Cardinality	Field
PRIMARY	PRIMARY	0	roomNo

USER DETAILS:

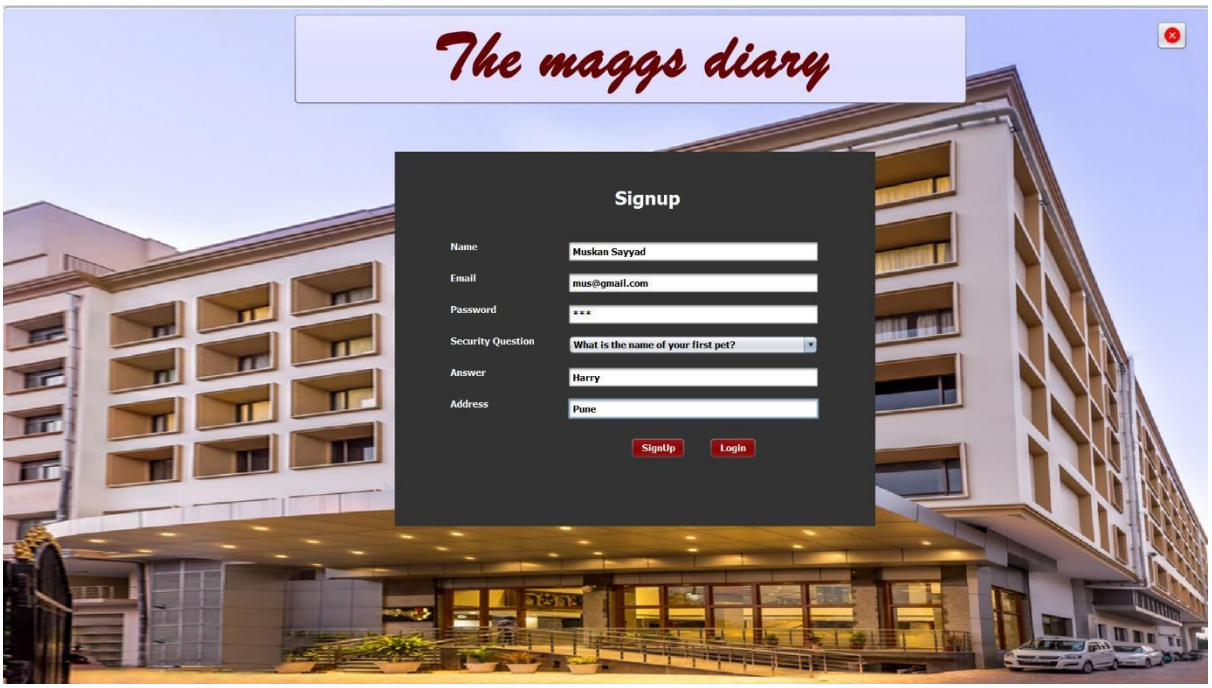
Field	Type	Null	Key	Default
name	Varchar(200)	YES		NULL
email	varchar(200)	YES		NULL
password	varchar (50)	YES		NULL
securityQuestion	Varchar(500)	YES		NULL
answer	Varchar(200)	YES		NULL
address	Varchar(200)	YES		NULL
status	varchar(20)	YES		NULL

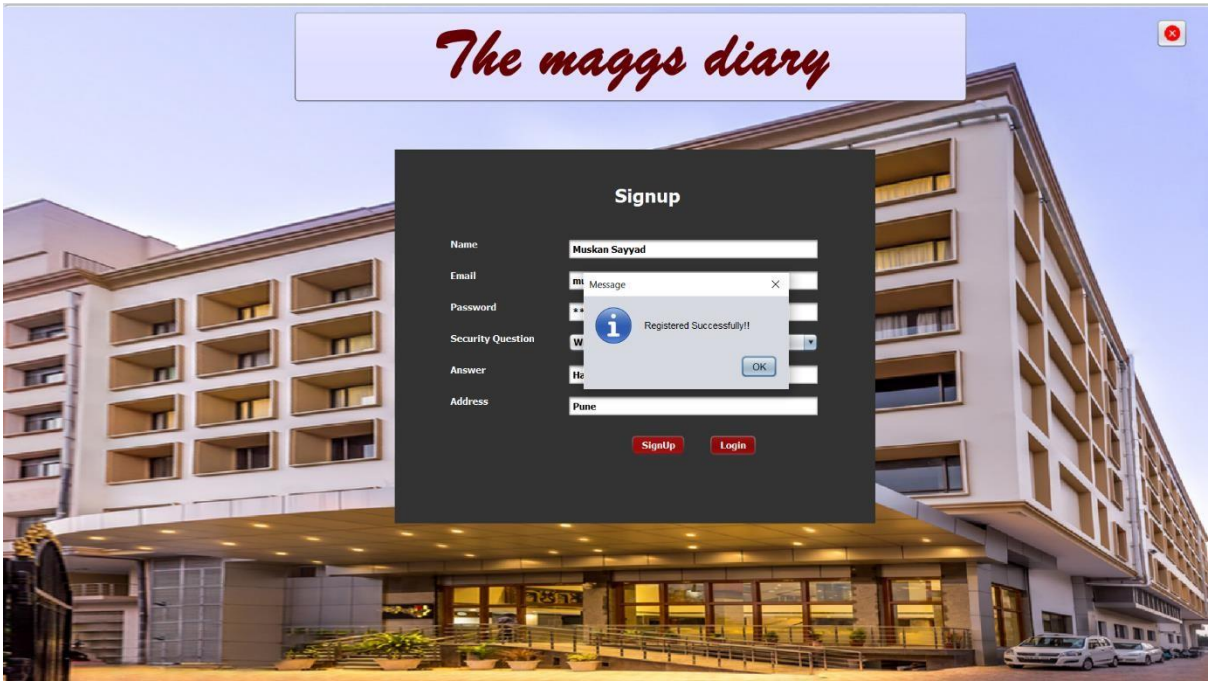
CUSTOMER DETAILS:

Field	Type	Null	Key	Default
id	int	YES		NULL
name	varchar(200)	YES		NULL
mobileNumber	varchar (20)	YES		NULL
nationality	Varchar(50)	YES		NULL
gender	Varchar(200)	YES		NULL
email	Varchar(200)	YES		NULL
idProof	Varchar(200)	YES		NULL
idNumber	Varchar(500)	YES		NULL
checkIn	Varchar(50)	YES		NULL
roomNo	Varchar(10)	YES		NULL
Bed	Varchar(200)	YES		NULL
roomType	Varchar(200)	YES		NULL
pricePerDay	int	YES		NULL
numberOfDaysStay	int	YES		NULL
totalAmount	Varchar(50)	YES		NULL
checkOut	varchar(50)	YES		NULL

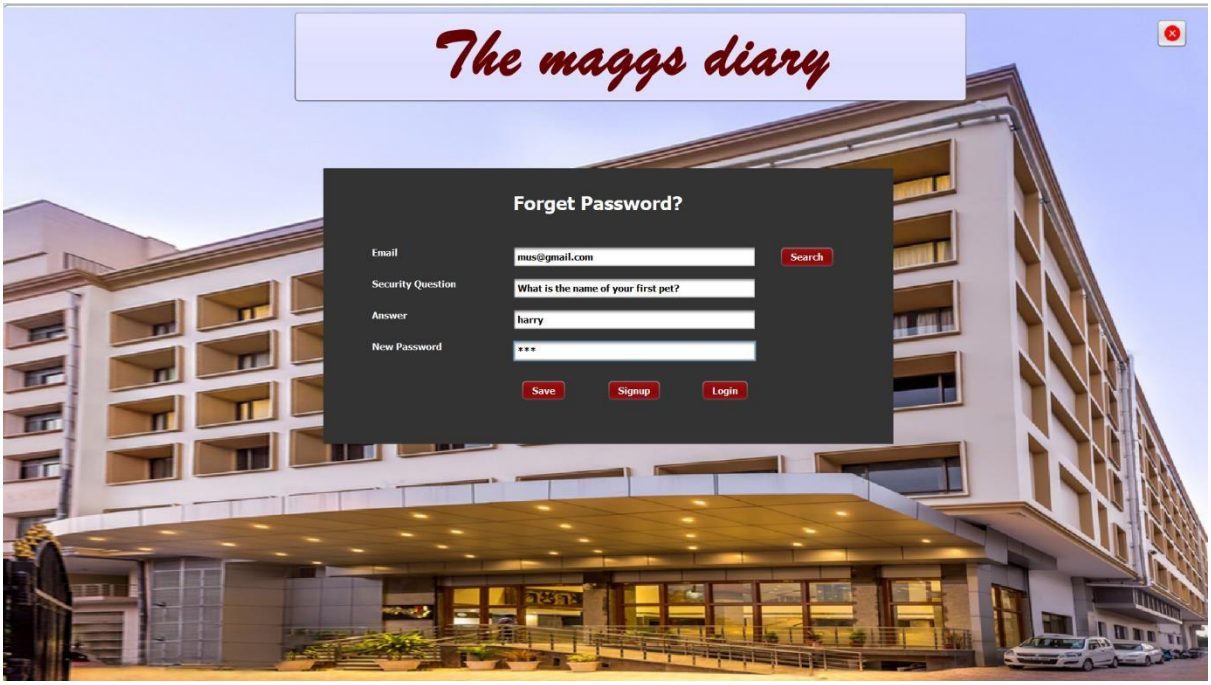
USER INTERFACE SCREEN

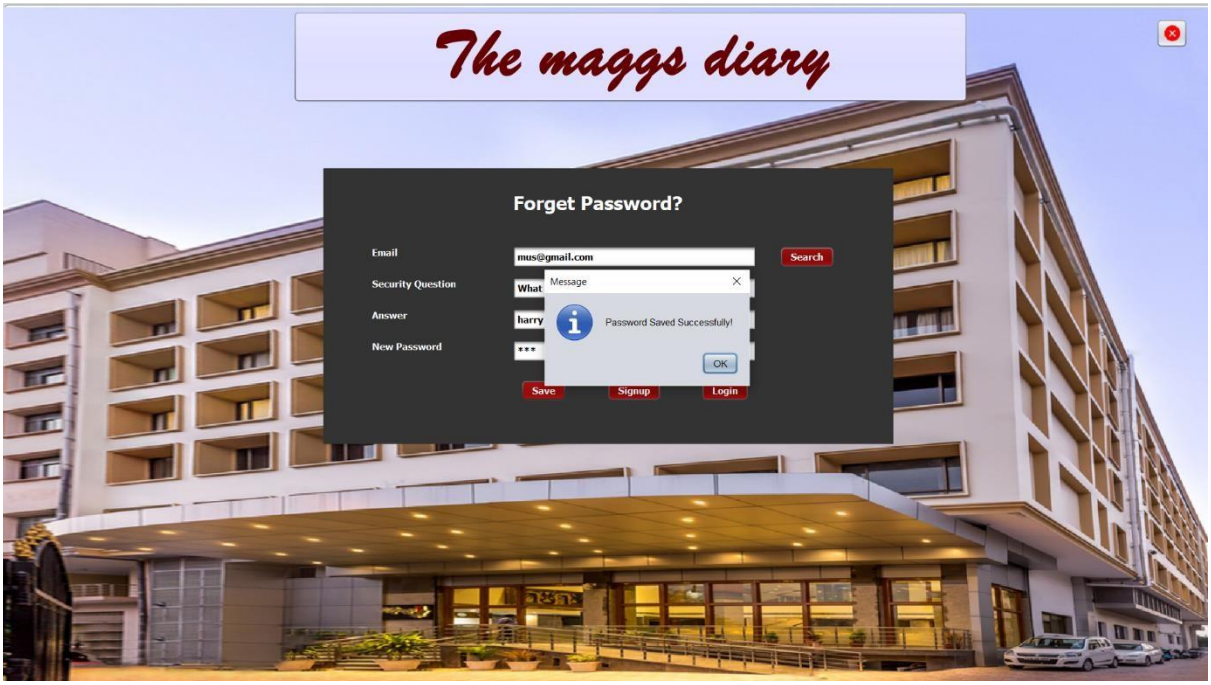
SingUp



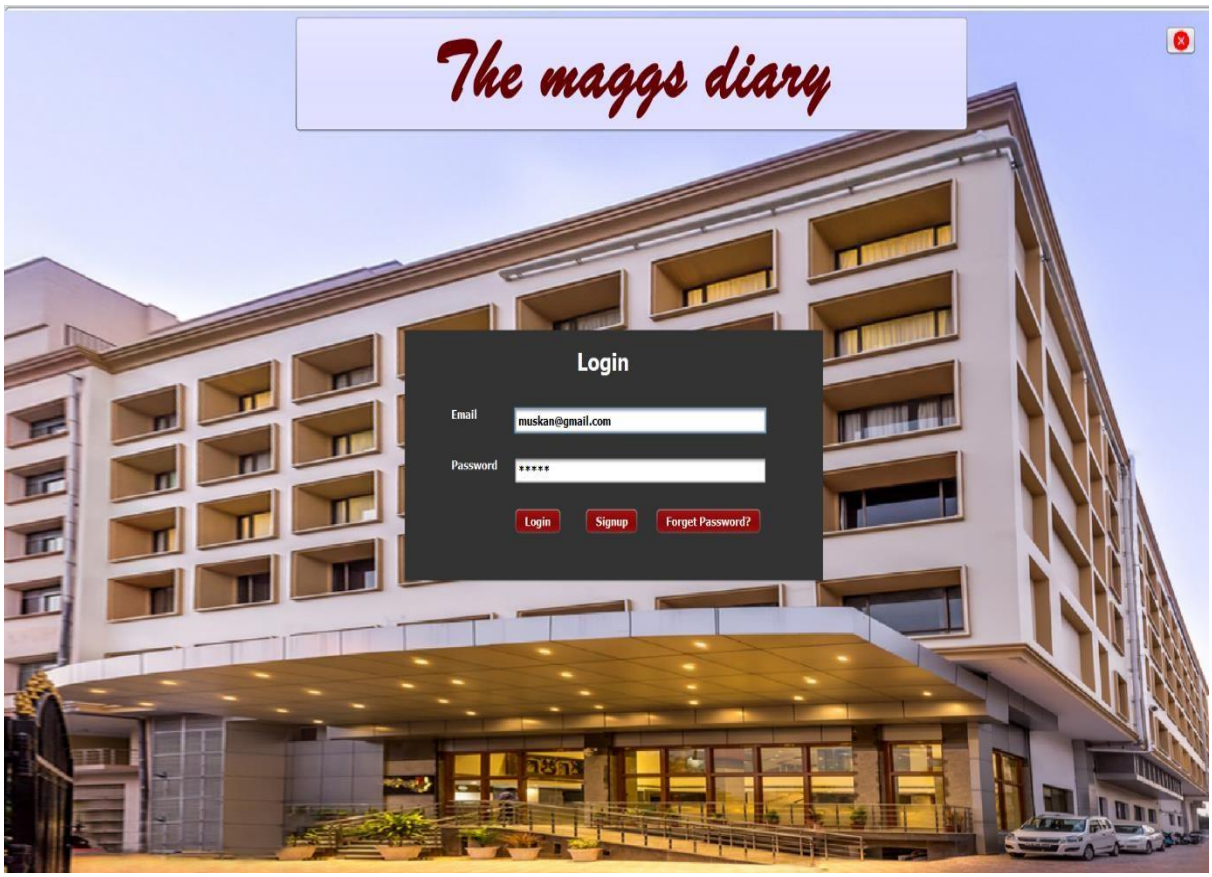


Forget Password

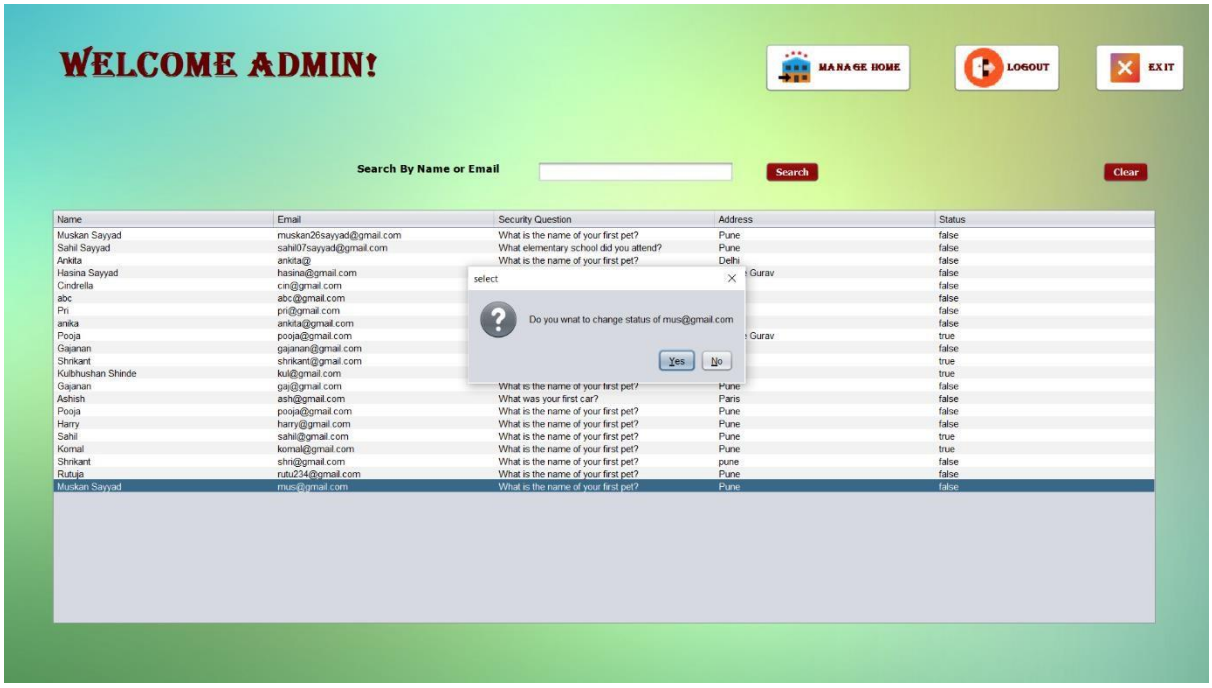


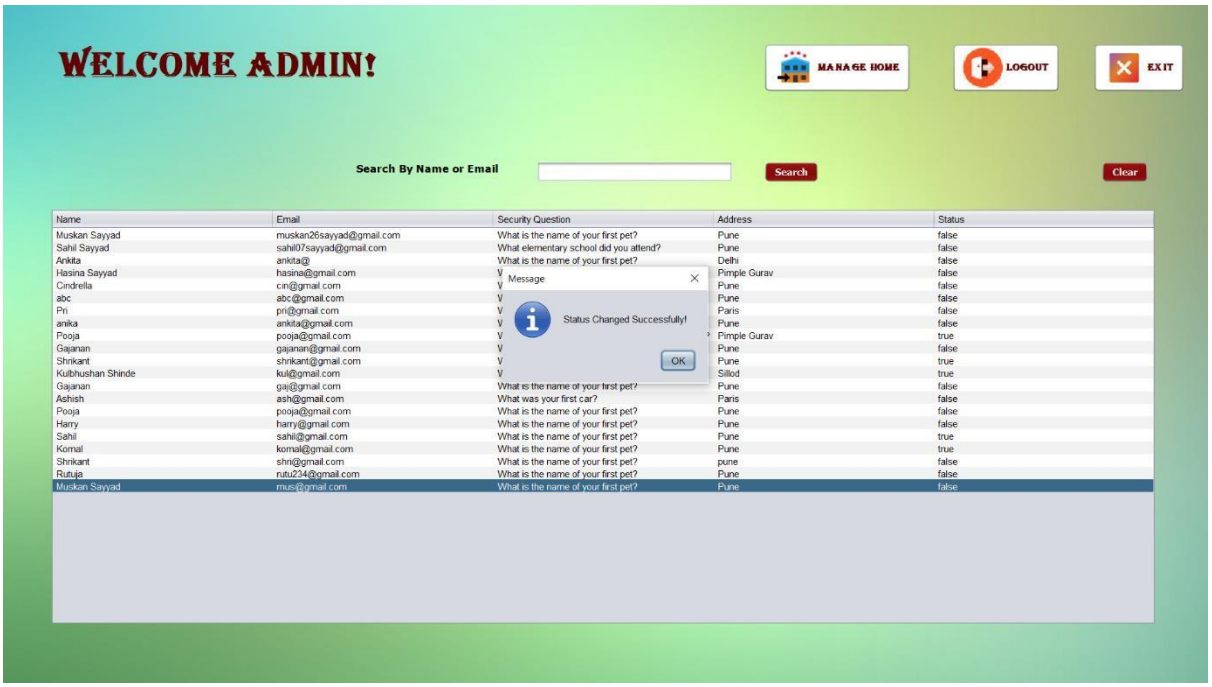


Admin Login

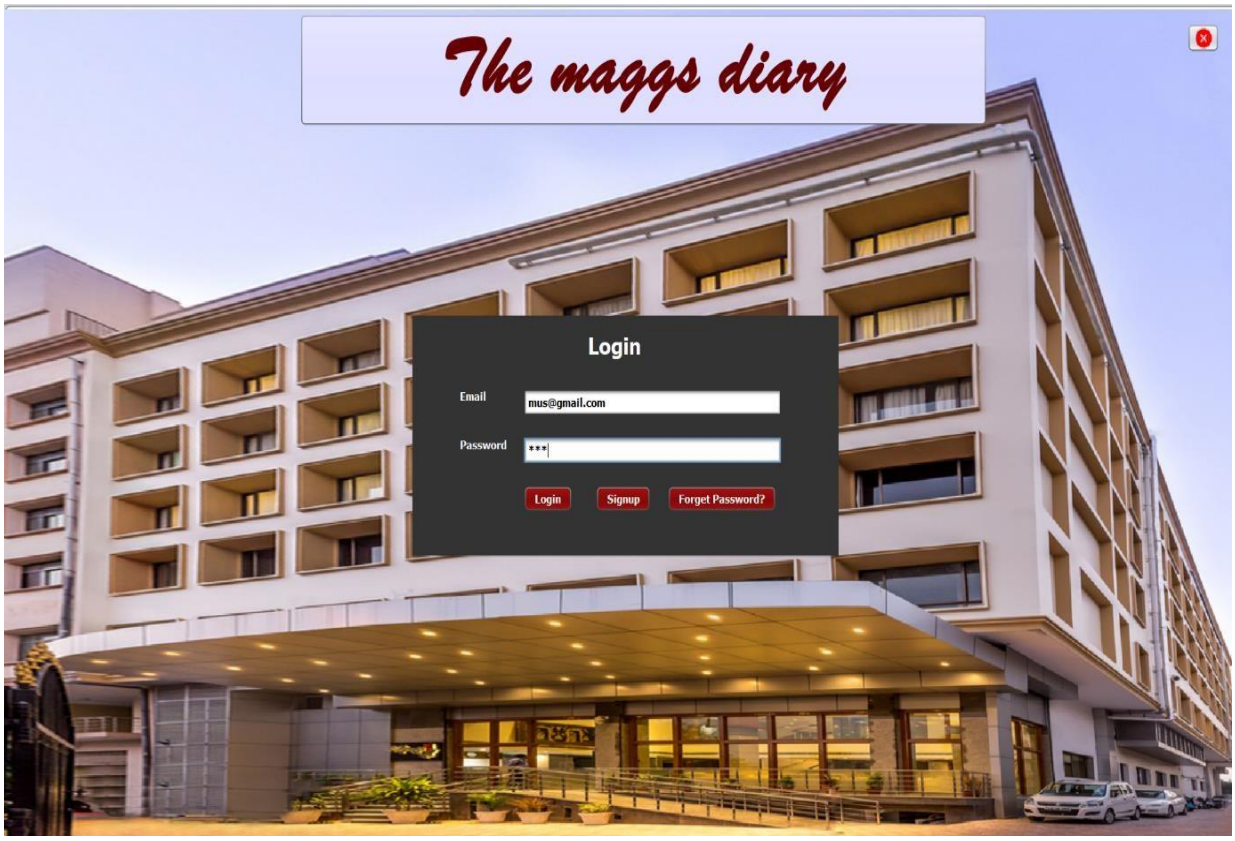

The maggs diary
Login
Email
muskan@gmail.com
Password
*****
Login
Signup
Forget Password?

Admin HOME

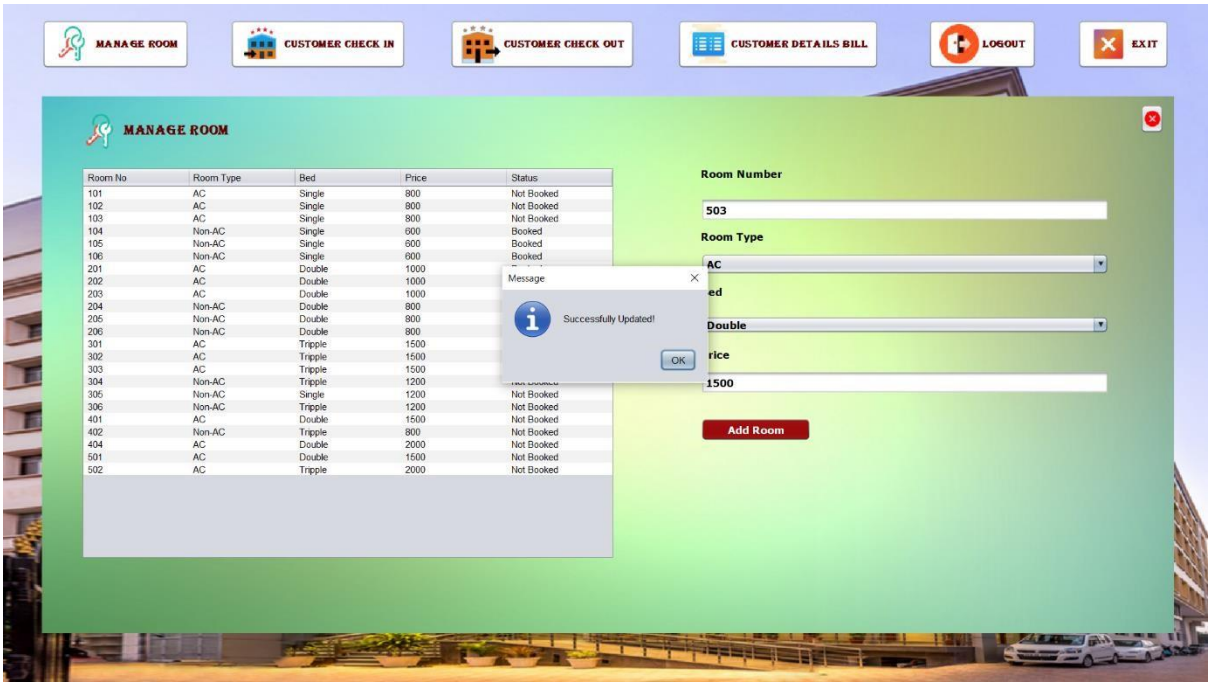

WELCOME ADMIN!
MANAGE HOME
LOGOUT
EXIT
Search By Name or Email
Search
Clear
Name
Email
Security Question
Address
Status
select
Do you wnat to change status of mus@gmail.com
Yes
No


WELCOME ADMIN!
MANAGE HOME
LOGOUT
EXIT
Search By Name or Email
Search
Clear
Name
Email
Security Question
Address
Status
Message
Status Changed Successfully!
OK

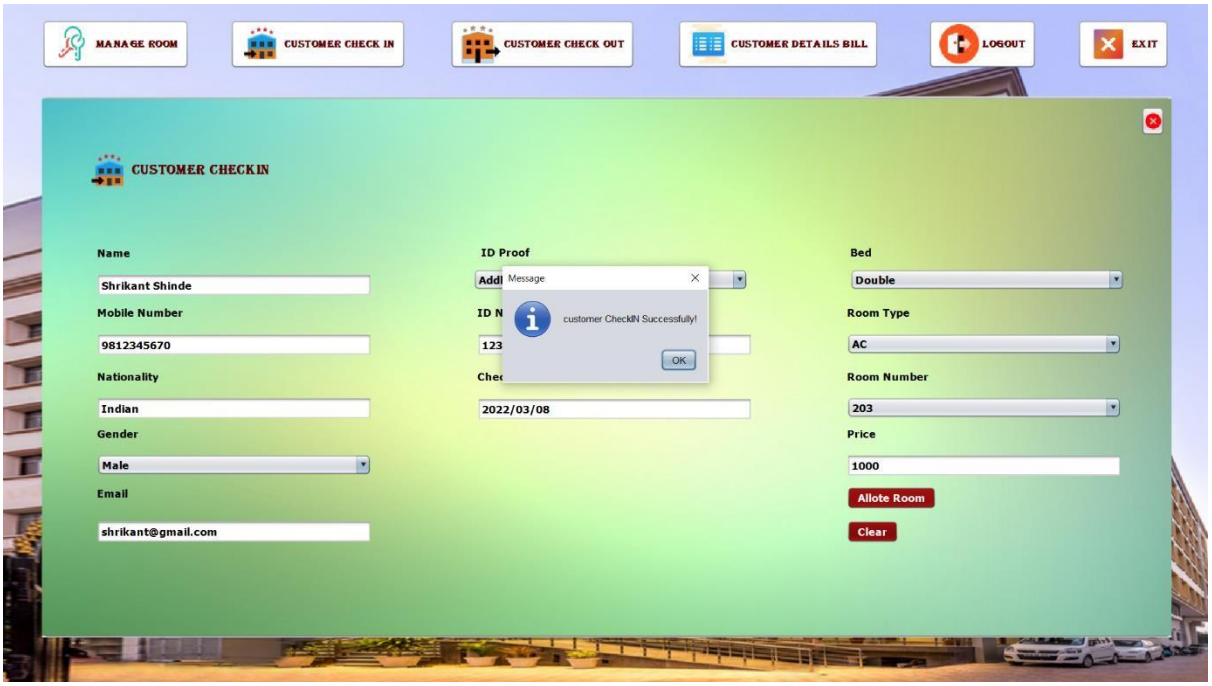
Staff Login


The maggs diary
Login
Email
mus@gmail.com
Password
Login
Signup
Forget Password?

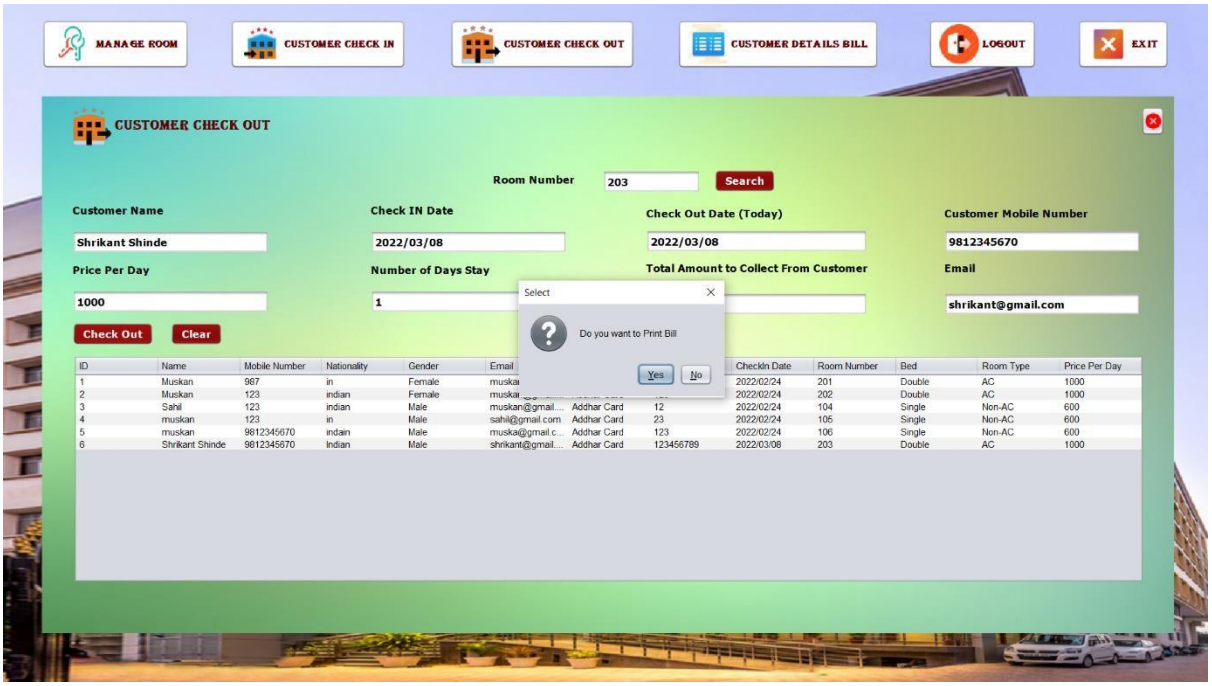
Manage Room



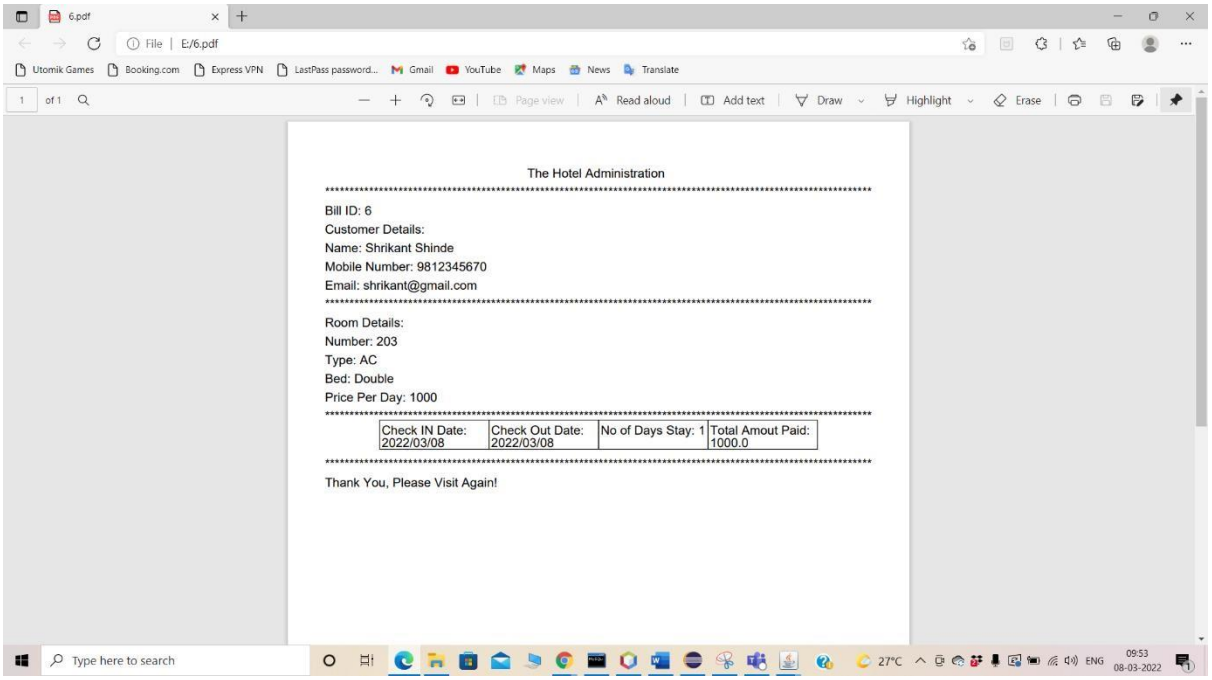
CheckIN



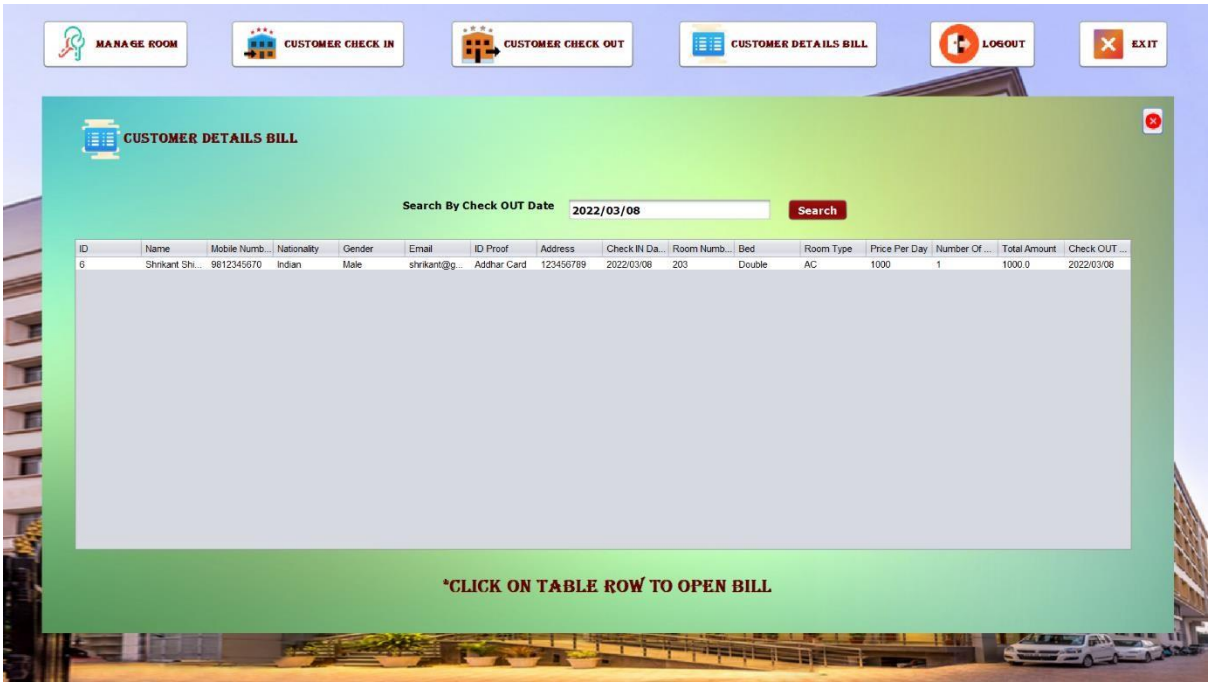
CheckOut



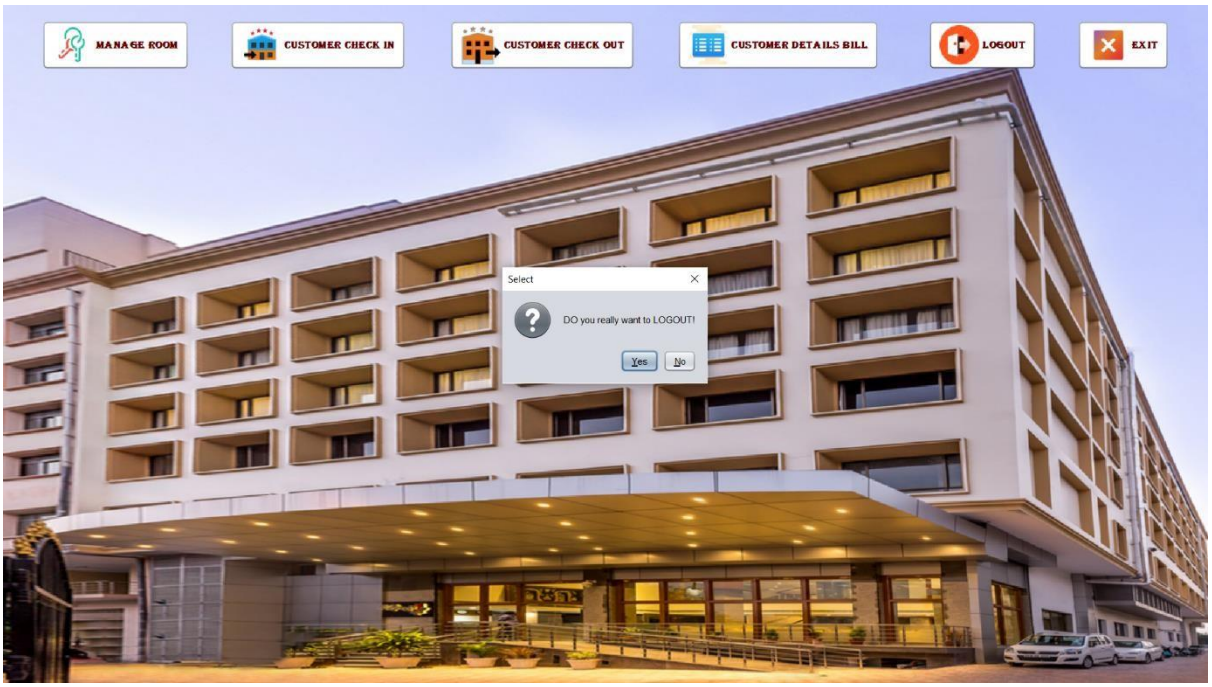
Print Bill



Billing Details



LogOut



VALIDATION CHECKS:

TEST CASE FOR ROOM RESERVATION:

- Test case id: TC_HS_001
- Test Case Name: Room Reservation
- Test Item: Room reservation form

Description: To test if the system successfully reserves room(s) as per user requirements

Procedure:
1. Go to the HMS home page
2. Select room reservation
3. Fill the required fields

Input 1: Enter information in all fields correctly
Expected Output 1: User successfully reserves a room

Input 2: Enter a room number that has already been reserved
Expected output 2: Generates a message proclaiming, "Room has already been reserved. Select another room"

Input 3: Fill any of the fields incorrectly. (E.g. input alphabets in the field requiring the room number)
Expected output 3: Generates an error message, pointing out the field in which error has occurred.

Input 4: Provide incomplete information (leave a few fields blank)
Expected output 4: Generates a message proclaiming, "Incomplete information provided".

TEST CASE FOR STAFF FORGET PASSWORD:

- Test case id: TC_HS_002
- Test Case Name: Forget Password
- Test Item: Forget Password form

Description: This test is designed to check if authorized staff members can recover their password into the designed system as per the outlined requirements.

Procedure:
1. Go to HMS home page
2. Select forget password
3. Fill the required fields

Input 1: Enter information in all fields correctly
Expected Output 1: User successfully recover the password.

Input 2: If staff enters incorrect email and security password
Expected Output 2: Generates an error message, pointing out the information is incorrect to proceed further.

TEST CASE FOR ADMIN AND STAFF LOG IN:

- Test case id: TC_HS_003
- Test Case Name: Room Reservation
- Test Item: Room reservation form

Description: This test is designed to check if authorized admin members can log into the designed system as per the outlined requirements.

Procedure:
1. Go to HMS home page
2. Select login
3. Enter username
4. Enter and verify password

Input 1: Enter correct username and password
Expected output 1: Successful log in Input

Input 2: Enter correct username and incorrect password
Expected output 2: access is denied. Message generated, displaying: “Incorrect password. Try again”.

Input 3: Enter correct password and incorrect username
Expected Output 3: Access is denied. Message generated, displaying: “Incorrect username. Try again”.

TEST CASE FOR ADMIN AND STAFF LOG OUT:

- Test ID: TC_HMS_004
- Test Case Name: admin log out
- Test Item: log out form

Description: This test case will check if the system successfully logs out the user as per requirement.

Procedure:
1. Go to the HMS homepage
2. If not already logged in, do so
3. Select the log out tab

Input: Click on the log out tab
Expected Output: User is logged out. Message proclaiming, “You have been logged out”, is displayed.

IMPLEMENTATION AND MAINTENANCE:

Project overview

This section provides an overview of the purpose, scope and objectives of the assigned project for which the plan has been written. The plan essentially outlines the project assumptions and constraints, a list of project deliverables, a summary of the project schedule and budget, and the plan for evolving the Project Management Plan.

Purpose

The purpose of the project is to design and implement Hotel Management System to ensure efficient running of all the events and tasks associated with successful running of a hotel.

The project would provide software based application and associated hardware. The customer would find it feasible to login from a remote location and gather information about hotel, check room and party packages. Reserve rooms, halls or tables and check menu without visiting the hotel. Admin can easily handle and edit menu, job vacancies, billing, duties of workers etc.

The stake holders in this project will find it absolutely convincing to implement the project and they will judge it by way of its performance and handling of routine work.

Some main characteristics of this project are:

- The project shall only aid to enhance the working system of the organization, and not, in any way be responsible for affecting business rules or policies of the organization in general or particular.
- The project shall be delivered to the organization as per the requirements and specifications provided by the user organization. Any amendment or change to the final
  design shall be taken up only after due consultation with a member of user organization not below the rank of an executive.
- The consideration for up-gradation of the project to include all hotels in the world under one umbrella has been excluded from the present development. However, the same may be incorporated by intelligent re-use of the project.
- The business needs to be satisfied by the project include all those as provided in the requirements and specifications from the organization. The overall aim is to enhance the existing management system by providing an online system for handling day to day activities of the hotel. It also includes refining and improving the existing structure of the organization with a view to improve the business strategies, goals, profits and image in the open market of competition.

Project Deliverables

The project deliverables include following:

- Feasibility report. Project will start once the report is final.
- Project team meetings to analyze the requirements, validate and finalize them and determine project prototype.
- Customer meetings to clarify the requirements with them and gather their requirements.
- Necessary hardware required dedicated for the software based application developed for the purpose e.g. receipt printer.
- First version of the project deployed for reviewing purposes and testing phase.
- Final project deployment on hotel computer.
- Technical support given for the purpose of training and to remove issues in running the system.

The project shall be handed over to the user organization at their premises and first time installation and integration of the project with existing system of the organization is the responsibility of the developing team. However, subsequent installations shall be charged separately and shall not be included in the budget of the project development

Evolution of the plan

- The project shall be developed starting at modular levels, incremental development and integration plan and testing at every stage of integration.
- The planned updates and progress of the project shall be submitted to the user organization in a weekly summary, electronically mailed to the representative of the organization as nominated. The details shall be discussed in person in the fortnightly meetings.
- The unplanned aspects relating to the project development shall be intimated to the user organization immediately on occurrence and may be discussed with the development team on required basis. However, it is the responsibility of the user organization to address the unplanned aspects and queries of the development team within 24 hrs of the intimation, failing which the development team shall not be responsible for any delay.
- The initial version of this plan shall be placed for configuration management of the development team and user organization before the commencement of the project and shall be considered final should both the development and user side agree to the details of the initial version of the plan.
- Subsequent changes to this plan shall only be allowed upon mutual agreement of both sides. However, any change affecting the development of the project shall be discussed in detail, reviewing various aspects and their implications on the development cost, schedule etc. This would require approval of executives from both

sides. Any change in plan once approved shall be entered in the original document with relevant details of amendments, and amended copy provided to the user organization in the subsequent week for information.

Roles and responsibilities

The development team shall comprise of the following individuals:

- Project director: Responsible for making available all required resources. He shall remain in close contact with user organization for clarity in statement of requirement and specification. He shall also be responsible for making finances available for the project as per the contract details with the user organization. He will make sure that the subordinate team comprising the developer, system analyst and risk analyst are available and present in the fortnightly meetings and any other discussion as deemed necessary.
- Project manager: He shall act as the middle man between the individual developers/analysts and the project director and shall be responsible for implementation and execution of assigned tasks as per schedule. Any interaction for purpose of clarity shall be addressed to project director and routed through the project manager.
- Software developer: The developer shall be responsible for developing the project application along with various interfaces. He shall be responsible for completion of work within the allotted time as instructed by the project director, and the timeline outlined in the feasibility study already carried out by system analyst. The evolutionary development of project and integration of modules after testing shall be carried out by the developer. He shall prepare the software for the application in line with the international standards and ethics, and ensure the source code is well documented for subsequent maintenance by the developing team. He shall aid system analyst in preparation of necessary documents related to the project. He shall also be responsible for submitting weekly reports regariding respective assignments and progress to project manager.
- Risk analyst: He shall be responsible for analysing the various aspects of the project including indentification, responsibiltities assessment and contigency planning. He shall also recommend ways to reduce the risks by suggesting alternate ways to contigency planning. He shall aid the system analyst in preparation of various documents related to the project for handing over to the user organization.
- System analyst: He shall be responsible for carrying out the feasibility study of the complete project based on the user requirement, resources and skills available with the developing team, and the market trends. System analyst shall also be responsible to monitor the development of the project from financial point of view and carry out reestimation of the project, should there be any change in the initially agreed-upon plan. He shall also be responsible for preparation of the user manual and necessary

documentation to be provided to the user organization related to the project. In consultation with the risk analyst and developer

Start-up plans

Estimation plan

- Specify the estimated cost, schedule and resource requirements for conducting the project and specify the associated confidence levels for each estimate.
- The estimated cost to be incurred on the project shall include the expenditure incurred on making the resources available for development, pays of development team assigned to develop and complete the project as well as miscellaneous administrative expenditures.
- The cost incurred on transportation and other allied administrative expenditures incurred on making the resources available for development shall contribute to the cost of resources.
- The estimates also include the pay of the development team for the duration of the project development as mutually agreed upon.
- The estimates shall also include various expenditures incurred on meetings, transportation, and support services as deemed feasible and necessary and required by the development team from time to time.
- The estimates of various aspects of the project shall be principally based on the estimates worked out in the feasibility study.
- Re-estimates shall be worked out based on the amendments to actual user requirements and their implications to the design and development of project. System analyst shall be responsible for carrying out detailed re-estimation and the re-estimates shall be shared with the user organization for necessary availability of funding.

Control plan

Requirements control

- User shall be cautioned for requirements change and their implications to the development of the project.
- Any change in requirement shall be scrutinized and analyzed in detail to assess its implications on project development as a whole and design and budgeting in specific. The change in requirements and their resulting impact and implications on the scope and objective of the project shall be deliberated with the user organization.
- Risk factors associated with the requirements change shall be identified and shared with the user organization.
- Any change recommended should strictly conform to specifications of change control process.

Schedule Control

- The project shall be closely monitored for measuring, reporting and controlling schedule of the project.

- The project manager shall hold a 15 min meeting daily at the start of the day and at the end of the day.
- The start of the day meeting shall be used to provide guidelines for the day's work and reiterate an aspect considered important.
- The end of the day meeting shall be used to confirm achieving of the day's targets and communicate faults, failures or shortfalls.
- These meetings shall be coordinated by the project manager.

Budget Control

- The budget shall be controlled by the project director and managed by the project manager. An end-of-week summary shall be prepared and put to the project director to intimate the weekly expenditures and required funds for subsequent week.
- Where the activities are exceeding the budget allocation the issue shall be addressed on priority to identify the variations and if required, user organization be intimated for additional grants.

Quality Control

- The entire project shall be developed in accordance with international quality standards.
- Strict ethic codes for software developers shall be followed in development of the project.
- The project shall be well documented and commented for ease of maintenance and reusability.

Reporting

- The staff involved in the development shall strictly follow the daily 15-min meeting for reporting and feedback.
- A weekly progress report shall be submitted to the project director by the project manager incorporating all aspects of development, including schedule, budget, resources etc.
- Where required the individuals shall communicate directly with the project manager on issue pertaining their domain.

Risk Management Plan

This plan processes procedures that will be followed to manage and control events that could have a negative impact on the Hotel Management System project.

It is the controlling document for managing all the project risks.

The risk management plan shall address following aspects of risk analysis:-
Risk identification

- Risk responsibilities

- Risk assessment
- Risk mitigation
- Risk contingency planning
- Risk tracking and reporting

Detailed aspects of risk management plan are elaborated

Project Close-out Plan

- The project shall be handed over to the user organization upon completion. The project deliverables shall include the system as well as necessary documentation.
- Following the completion and handing over, the staff shall be reassigned to subsequent project.
- The necessary project material shall be archived in the team's business project database.
- An internal in house debriefing shall be carried out, where the team individuals shall share their personnel experiences and provide valuable lessons learnt from the project development.

SAMPLE PROGRAM CODE:

- **connectionProvider.java**

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.*;

public class ConnectionProvider {
   public static Connection getCon()
   {
      try
      {
         Class.forName("com.mysql.cj.jdbc.Driver");
         Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/hotel","root","root")
;
         return con;
      }
      catch(Exception e)
      {
         return null;
      }
   }
}
```

- **forgetPassword.java**

```
import javax.swing.JOptionPane;
import java.sql.*;
import project.*;
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to edit this template
 */

/**
 *
 * @author muska
 */
public class forgetPassword extends javax.swing.JFrame {

    /**
     * Creates new form forgetPassword
     */
    public forgetPassword() {
        initComponents();
    }
    String email;

    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {
```

```
        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jLabel3 = new javax.swing.JLabel();
        jButton1 = new javax.swing.JButton();
        jLabel4 = new javax.swing.JLabel();
        jLabel5 = new javax.swing.JLabel();
        jTextField1 = new javax.swing.JTextField();
        jTextField2 = new javax.swing.JTextField();
        jTextField3 = new javax.swing.JTextField();
        jPasswordField1 = new javax.swing.JPasswordField();
        jButton2 = new javax.swing.JButton();
        jButton3 = new javax.swing.JButton();
        jButton6 = new javax.swing.JButton();
        jButton4 = new javax.swing.JButton();
        jButton5 = new javax.swing.JButton();
        jButton7 = new javax.swing.JButton();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        setUndecorated(true);
        getContentPane().setLayout(new org.netbeans.lib.awtextra.AbsoluteLayout());

        jLabel1.setFont(new java.awt.Font("Tahoma", 1, 24)); // NOI18N
        jLabel1.setForeground(new java.awt.Color(255, 255, 255));
        jLabel1.setText("Forget Password?");
        getContentPane().add(jLabel1, new
org.netbeans.lib.awtextra.AbsoluteConstraints(650, 240, 231, -1));

        jLabel2.setFont(new java.awt.Font("Tahoma", 1, 12)); // NOI18N
```

```
jLabel2.setForeground(new java.awt.Color(255, 255, 255));
        jLabel2.setText("Email");
        getContentPane().add(jLabel2, new
org.netbeans.lib.awtextra.AbsoluteConstraints(470, 310, 37, -1));

        jLabel3.setFont(new java.awt.Font("Tahoma", 1, 12)); // NOI18N
        jLabel3.setForeground(new java.awt.Color(255, 255, 255));
        jLabel3.setText("Security Question");
        getContentPane().add(jLabel3, new
org.netbeans.lib.awtextra.AbsoluteConstraints(470, 350, -1, -1));

        jButton1.setBackground(new java.awt.Color(102, 0, 0));
        jButton1.setFont(new java.awt.Font("Tahoma", 1, 12)); // NOI18N
        jButton1.setForeground(new java.awt.Color(255, 255, 255));
        jButton1.setText("Save");
        jButton1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton1ActionPerformed(evt);
            }
        });
        getContentPane().add(jButton1, new
org.netbeans.lib.awtextra.AbsoluteConstraints(660, 480, -1, -1));

        jLabel4.setFont(new java.awt.Font("Tahoma", 1, 12)); // NOI18N
        jLabel4.setForeground(new java.awt.Color(255, 255, 255));
        jLabel4.setText("Answer");
        getContentPane().add(jLabel4, new
org.netbeans.lib.awtextra.AbsoluteConstraints(470, 390, -1, -1));
```

```
jLabel5.setFont(new java.awt.Font("Tahoma", 1, 12)); // NOI18N
        jLabel5.setForeground(new java.awt.Color(255, 255, 255));
        jLabel5.setText("New Password");
        getContentPane().add(jLabel5, new
org.netbeans.lib.awtextra.AbsoluteConstraints(470, 430, -1, -1));

        jTextField1.setFont(new java.awt.Font("Tahoma", 1, 12)); // NOI18N
        jTextField1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jTextField1ActionPerformed(evt);
            }
        });
        getContentPane().add(jTextField1, new
org.netbeans.lib.awtextra.AbsoluteConstraints(650, 310, 310, -1));

        jTextField2.setFont(new java.awt.Font("Tahoma", 1, 12)); // NOI18N
        getContentPane().add(jTextField2, new
org.netbeans.lib.awtextra.AbsoluteConstraints(650, 350, 310, -1));

        jTextField3.setFont(new java.awt.Font("Tahoma", 1, 12)); // NOI18N
        getContentPane().add(jTextField3, new
org.netbeans.lib.awtextra.AbsoluteConstraints(650, 390, 310, -1));

        jPasswordField1.setFont(new java.awt.Font("Tahoma", 1, 12)); // NOI18N
        getContentPane().add(jPasswordField1, new
org.netbeans.lib.awtextra.AbsoluteConstraints(650, 430, 310, -1));

        jButton2.setBackground(new java.awt.Color(102, 0, 0));
        jButton2.setFont(new java.awt.Font("Tahoma", 1, 12)); // NOI18N
        jButton2.setForeground(new java.awt.Color(255, 255, 255));
```

```
jButton2.setText("Signup");
        jButton2.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton2ActionPerformed(evt);
            }
        });
        getContentPane().add(jButton2, new
org.netbeans.lib.awtextra.AbsoluteConstraints(770, 480, -1, -1));

        jButton3.setBackground(new java.awt.Color(102, 0, 0));
        jButton3.setFont(new java.awt.Font("Tahoma", 1, 12)); // NOI18N
        jButton3.setForeground(new java.awt.Color(255, 255, 255));
        jButton3.setText("Login");
        jButton3.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton3ActionPerformed(evt);
            }
        });
        getContentPane().add(jButton3, new
org.netbeans.lib.awtextra.AbsoluteConstraints(890, 480, -1, -1));

        jButton6.setBackground(new java.awt.Color(102, 0, 0));
        jButton6.setFont(new java.awt.Font("Tahoma", 1, 12)); // NOI18N
        jButton6.setForeground(new java.awt.Color(255, 255, 255));
        jButton6.setText("Search");
        jButton6.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton6ActionPerformed(evt);
            }
```

```
});
        getContentPane().add(jButton6, new
org.netbeans.lib.awtextra.AbsoluteConstraints(990, 310, -1, -1));

        jButton4.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/images/close.png"))); // NOI18N
        jButton4.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton4ActionPerformed(evt);
            }
        });
        getContentPane().add(jButton4, new
org.netbeans.lib.awtextra.AbsoluteConstraints(1470, 20, 40, -1));

        jButton5.setBackground(new java.awt.Color(204, 204, 255));
        jButton5.setFont(new java.awt.Font("Brush Script MT", 1, 85)); // NOI18N
        jButton5.setForeground(new java.awt.Color(102, 0, 0));
        jButton5.setText("The maggs diary");
        getContentPane().add(jButton5, new
org.netbeans.lib.awtextra.AbsoluteConstraints(370, 10, 860, -1));

        jButton7.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/images/forgot password.png"))); //
NOI18N
        getContentPane().add(jButton7, new
org.netbeans.lib.awtextra.AbsoluteConstraints(0, 0, 1550, -1));

        pack();
    }// </editor-fold>
```

```
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    setVisible(false);
    new login().setVisible(true);
  }

  private void jTextField1ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
  }

  private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    int a=JOptionPane.showConfirmDialog(null,"Do you really want to close this
Application?","Select",JOptionPane.YES_NO_OPTION);
    if(a==0)
      System.exit(0);
  }

  private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    int check=0;
    String securityQuestion=jTextField2.getText();
    String answer=jTextField3.getText();
    String newPassword = jPasswordField1.getText();
    if(answer.equals("") || newPassword.equals(""))
    {
      check=1;
      JOptionPane.showMessageDialog(null, "All Fields Is Required!");
```

```
}
      else
      {
         ResultSet rs = Select.getData("select *from users where email = '"+email+"' and
securityQuestion= '"+securityQuestion+"' and answer='"+answer+"'");

         try
         {
            if(rs.next())
            {
               check=1;
               InsertUpdateDelete.setData("update users set password
='"+newPassword+"' where email='"+email+"'","Password Saved Successfully!");
               setVisible(false);
               new forgetPassword().setVisible(true);
            }
         }
         catch(Exception e)
         {
            JOptionPane.showMessageDialog(null, e);
         }
      }
      if(check==0)
      {
         JOptionPane.showMessageDialog(null, "Incorrect Answer!");
      }
   }

   private void jButton6ActionPerformed(java.awt.event.ActionEvent evt) {
```

```
// TODO add your handling code here:
    int check=0;
    email=jTextField1.getText();
    if(email.equals(""))
    {
       check=1;
       JOptionPane.showMessageDialog(null, "Email Field Is Required!");
    }
    else
    {
       ResultSet rs = Select.getData("select *from users where email='"+email+"'");
       try
       {
          if(rs.next())
          {
             check=1;
             jTextField2.setEditable(false);
             jTextField1.setEditable(false);
             jTextField2.setText(rs.getString(4));
          }
       }
       catch(Exception e)
       {
          JOptionPane.showMessageDialog(null, e);
       }
    }
    if(check==0)
    {
```

```
JOptionPane.showMessageDialog(null, "Incorrect Email!");
        }
    }

    private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        setVisible(false);
        new signup().setVisible(true);
    }

    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
        /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
         * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
         */
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {
                    javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
```

```
}
        } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(forgetPassword.class.getName()).log(java.util.loggin
g.Level.SEVERE, null, ex);
        } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(forgetPassword.class.getName()).log(java.util.loggin
g.Level.SEVERE, null, ex);
        } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(forgetPassword.class.getName()).log(java.util.loggin
g.Level.SEVERE, null, ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(forgetPassword.class.getName()).log(java.util.loggin
g.Level.SEVERE, null, ex);
        }
        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new forgetPassword().setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify
    private javax.swing.JButton jButton1;
    private javax.swing.JButton jButton2;
```

```
    private javax.swing.JButton jButton3;
    private javax.swing.JButton jButton4;
    private javax.swing.JButton jButton5;
    private javax.swing.JButton jButton6;
    private javax.swing.JButton jButton7;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel3;
    private javax.swing.JLabel jLabel4;
    private javax.swing.JLabel jLabel5;
    private javax.swing.JPasswordField jPasswordField1;
    private javax.swing.JTextField jTextField1;
    private javax.swing.JTextField jTextField2;
    private javax.swing.JTextField jTextField3;
    // End of variables declaration
}
```

Validations:

- **customerCheckIn.java**

```
String regex="^[A-Za-z0-9+_.-]+@(.+)$";
String regex2="(0/91)?[7-9][0-9]{9}";
private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    int id=1;
    String name=jTextField1.getText();
    String mobileNumber=jTextField2.getText();
    String nationality=jTextField3.getText();
```

```
String gender=(String)jComboBox1.getSelectedItem();
String email=jTextField4.getText();
String idProof=(String)jComboBox5.getSelectedItem();
String idNo=jTextField6.getText();
String checkIN=jTextField7.getText();
String bed=(String)jComboBox2.getSelectedItem();
String roomType=(String)jComboBox3.getSelectedItem();
String roomNo=(String)jComboBox4.getSelectedItem();
String price=jTextField8.getText();

String Query="select max(id) from customer";
Pattern pattern = Pattern.compile(regex);
Matcher matcher = pattern.matcher(email);
Boolean match = matcher.matches();

Pattern pattern2 = Pattern.compile(regex2);
Matcher matcher2 = pattern2.matcher(mobileNumber);
Boolean match2 = matcher2.matches();

ResultSet rs = Select.getData("select *from customer where email='"+email+"'");

try
{
   if(match.equals(false)){
      JOptionPane.showMessageDialog(null,"Please provide valid email address!");
   }
   else if(name.equals("") || email.equals("") || mobileNumber.equals("") ||
nationality.equals("") || gender.equals("") || idProof.equals("") || idNo.equals("") ||
```

```
checkIN.equals("") || bed.equals("") || roomType.equals("")|| roomNo.equals("")|| price.equals(""))
            JOptionPane.showMessageDialog(null,"Every Field Is Required!!");
        else if(rs.next()){
            JOptionPane.showMessageDialog(null, "This email is already Registered!!");
            rs.close();
        }
         else if(match2.equals(false)){
            JOptionPane.showMessageDialog(null,"Please provide valid Mobile Number!");
        }

        else{
            rs=Select.getData(Query);
            while(rs.next())
                id=rs.getInt(1);
            id=id+1;

            if(!price.equals(""))
            {
                Query="update room set status='Booked' where roomNo='"+roomNo+"'";
                InsertUpdateDelete.setData(Query,"");
                Query="insert into customer(id,name,mobileNumber,nationality,gender,email,idProof,idNumber,checkIN,roomNo,bed,roomType,pricePerDay)
values("+id+",'"+name+"','"+mobileNumber+"','"+nationality+"','"+gender+"','"+email+"','"+idProof+"','"+idNo+"','"+checkIN+"','"+roomNo+"','"+bed+"','"+roomType+"','"+price+"')";
                InsertUpdateDelete.setData(Query, "customer CheckIN Successfully!");
                setVisible(false);
                new CustomerCheckIn().setVisible(true);
            }
```

```
}
        }
        catch(Exception e)
        {
           JOptionPane.showMessageDialog(null, e);
        }
     }
```

FUTURE SCOPE OF THE MINI PROJECT:

The project can be used in the hotel after some more useful modules in the project for which hotel are providing services.

Ulmost care and back-up procedures must be established to ensure 100% successful implementation of the computerized hotel system. In case of system failure, the organization should be in a position to process the transaction with another organization or if the worst comes to the worst, it should be in a position to complete in manually.

BIBLIOGRAPHY/REFERENCES/GLOSSARY:

- "Leam Java in 21 days "by SAM's publication
- Software Engineering by Roger s. Pressman
- Herbert Schildt "Java 2-The Complete Reference

- https://www.tutorialspoint.com/index.htm
- https://www.javatpoint.com
- https://www.w3school.com